

# Capitulo 2: MatrIx operatIons

## Resumen de capitulo

De manera muy resumida en este capitulo repasamos la forma de operar con las clases:

* BiPoarUtil
* Matrix
* MatrixMath

Considero que este conjunto de clases tiene sentido solo sin son utilizadas en conjunto, dado que proveen un conjunto de métodos y clases que dan soporte a un mismo fin, manejo de vectores y matrices. Con el fin de facilitar la manipulación y dejar el código mas limpio y manejable a la hora de hablar de redes neuronales artificiales.

Primero tenemos la clase *Matrix,* esta clase tiene el único fin de construir el objeto de tipo Matrix, el cual, representa una matrizy nos provee varios métodos para su manejo y manipulación.

La clase MatrixMath utiliza la clase Matrix. Esta clase provee un conjunto de métodos que pueden operar sobre las matrices:

* Suma
* Resta
* Multiplicación
* División
* Producto punto
* Traspuesta
* Matriz identidad
* Magnitud

Y cada uno de esto métodos recibe como parámetro(s) objeto(s) del tipo Matrix.

Por último, tenemos la clase BiPolarUtil, posee solo dos métodos (bipolar2double y double2bipolar). Pero esta clase nos da soporte para las redes neuronales que funcionan con valores booleanos, permitido que realizar operaciones sobre ellas. Su finalidad radica en la conversión de los valores booleanos:

True = 1

False = -1

## Preguntas de revisión

1. **¿Cuál es el propósito de usar números bipolares, en lugar de números booleanos?**

Las redes neuronales requieres que estén representados de con numero bipolares para poder realizar sus operaciones de manera correcta.

1. **¿La multiplicación de matrices es conmutativa?**

No, dado que si tenemos dos matrices con dimensiones:

“2x3” y “3x2”

Al realizar la multiplicación:

*Matriz “2x3” por Matriz “3x2”*

Obtendremos como resultado una matriz de dimensiones 2x2.

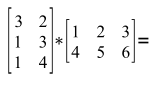
Si realizamos la multiplicación

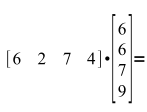
*Matriz “3x2” por Matriz “2x3”*

Obtendremos como resultado una matriz de dimensiones 3x3.

1. **¿Cuáles son las dimensiones de una matriz de peso utilizada para conectar una capa de dos neuronas a una capa de tres neuronas?**

De 3x2

1. **Realice la siguiente multiplicación.**

1. **Realice el siguiente producto de puntos.**

( 6 \* 6 ) + ( 2 \* 6 ) + ( 7 \* 7 ) + ( 4\*9 )= 133

## Vocabulario

**Bipolar:**

Conjunto de números que sustituyen a los valores booleanos dentro de los cálculos en redes neuronales, las equivalencias de los valores son:

True = 1

False = -1

**Boolean:**

Tipo de valor primitivo de java representado por un bit (0 ó 1) que representa true o false.

**Column Matrix:**

Representa una columna de una matriz dada. En la clase Matrix es un componente distinto al de row Matrix, aunque simbólicamente representan los mismo (un vector).

**Dot Product:**

Método estático de la clase MarixMath. Este método se encarga de obtener el producto punto de dos vectores. Recibe como argumento dos objetos del tipo Matirz.

**Identity Matrix:**

Matriz que tiene la propiedad de que al se multiplicada por una matriz esta no sufre ninguna alteración.

**Matrix:**

Clase que contiene todos los métodos necesarios para crear un array en un objeto de esta clase. Hablamos de matrices y vectores.

**Row Matrix:**

simbólicamente representa lo mismo que Col Marix (un vector), pero dentro de la clase Matrix se diferencia de este, mientras que ColMatrix representa una columna RowMatrix representa un renglón.

**Scalar:**

Valor numérico que no posee dimensiones. Solo representa un valor numérico.

**Vector:**

Se denomina vector a un conjunto de valores almacenados de manera continuo, generalmente del mismo tipo de valor. Para acceder a los datos dentro del vector es necesario iterar en el o conocer el índice donde esta almacenado el valor.

**Weight Matrix:**

Matriz que almacena los pesos de conexión entre las capas de la red neuronal.